

R&D 100 Awards

Mini-Apps Accelerate Hardware and Software Development

WHEN researchers and developers evaluate new high-performance computing (HPC) hardware approaches, especially radical departures from existing designs, a crucial step is selecting the right workloads to use in testing. During early stages of experimentation, they will often rely on small code fragments called kernels or benchmarking tools that offer streamlined deployment and data collection. Full-scale science and engineering applications, while the ultimate performance indicators, are often too large and complex to work with directly in early design studies. Thus, they are usually not executed until the final development stages, after the important design decisions are made. Unfortunately, kernels and benchmarking tools can be inadequate substitutes for real applications. In fact, as computer programs and

platforms grow more complex, scientific application developers have noted a widening gap between benchmark results and how their codes perform on fully realized HPC systems.

Occupying a desirable middle ground between benchmarks and full-scale applications is a new category of compact, self-contained proxies for real applications called mini-apps. The mini-app concept was pioneered by the Mantevo Project, which is led by Michael Heroux and Richard Barrett of Sandia National Laboratories. Mantevo Project contributors include Livermore computational physicists David Richards and James Belak, as well as researchers from Los Alamos and Sandia national laboratories, NVIDIA Corporation, and three British institutions—the University of Bristol, the University of Warwick, and the Atomic Weapons Establishment. Mantevo Suite 1.0, the first integrated collection of mini-apps, has earned an R&D 100 Award.

Identifying Performance Patterns

Mantevo mini-apps have quickly found a place in the HPC designer tool set because they can reliably predict application performance without the time, effort, and expertise that porting most full-scale codes to a new system would require. Averaging just 5,000 lines of code, the Mantevo mini-apps are comparatively easy to understand, deploy, modify, and even rewrite. As a consequence, they allow hardware developers to make earlier and better-informed design decisions, greatly improving the likelihood that the full-scale applications will perform successfully on the new system. Mini-apps are also useful for testing new programming models and languages, code compilation techniques, application scaling approaches, and other software-related experiments.

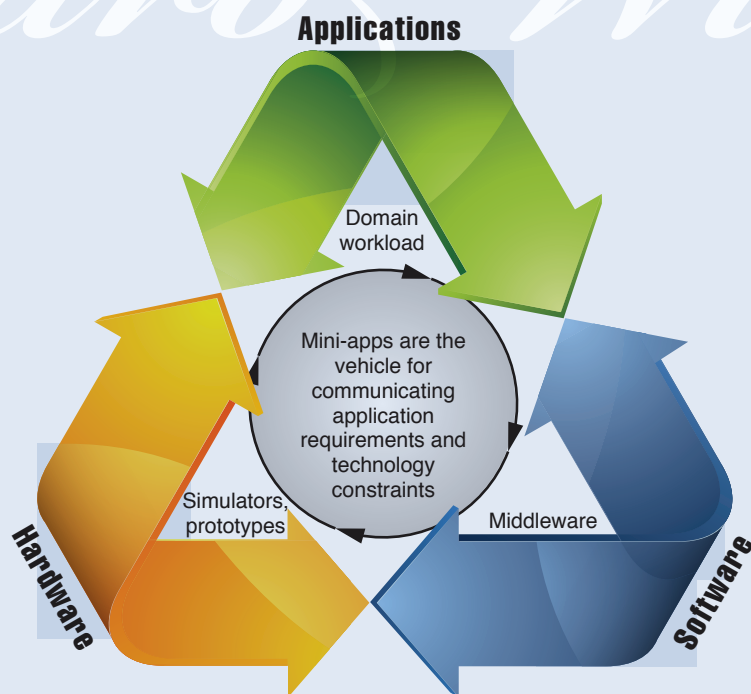
Mantevo Suite 1.0 contains miniature versions of seven HPC codes used for scientific and national security research. While many of the full-scale versions have access restrictions, the mini-apps are open-source software. Each mini-app is developed by the original application's code team, with the assistance of other experts. Although they are created independently and model different types of problems, all of the mini-apps share a common look and feel and are compatible with the suite's data collection and processing tools.

Mantevo mini-app developers strip a code of its nonessential elements and options and then focus on pattern identification. Although an application may have many modeling capabilities, the same underlying templates for data movement and processing



Livermore codevelopers of Mantevo Suite 1.0: (from left) James Belak and David Richards.

and Winner



will be repeated many times. By replicating only the essential patterns, the developers distill the essence of the full-sized program into a code a fraction of the program's size.

Livermore's contribution to the Mantevo Suite is the materials science proxy app Co-designed Molecular Dynamics (CoMD). Because CoMD contains most of the patterns in a full-sized classical MD code, it performs much the same. Furthermore, transformations that change how CoMD behaves will also apply to applications that share similar data patterns. Mantevo mini-apps such as CoMD are used regularly for performance design studies.

Predicting the Future by Inventing It

Computer scientists predict that the next generation of supercomputers will be notably different from previous generations. For the past two decades, hardware developers have produced faster computer microprocessors by increasing the speed and reducing the size of transistors, but transistors are nearing a lower limit in size and an upper limit in speed. Further speeding up the millions of transistors on a typical microprocessor would drive energy demands and operational costs for large-scale computing to untenable levels. (See *S&TR*, July/August 2013, pp. 4–13.) Going forward, performance gains will likely stem from redesigned processors that complete more computational tasks in parallel. Because the cost of data movement is also increasing, application developers are being asked to modify their algorithms. Instead of moving data to the compute processor, the computational work is being moved to the data.

These changes necessitate greater coordination between domain scientists, software developers, and hardware developers

Application-driven co-design brings together vendors, hardware architects, system software developers, domain scientists, computer scientists, and applied mathematicians to make informed decisions about features and trade-offs in the design of high-performance computing hardware, software, applications, and underlying algorithms. Proxy applications, or mini-apps, help communicate the computational workload of full-scale applications as part of the co-design process.

through an iterative and cooperative process called application-driven co-design. With application-driven co-design, scientific problem requirements influence computer architecture design, and system architecture technology constraints inform algorithm and software development. Lawrence Livermore researchers participate in several co-design projects funded by the Department of Energy (DOE), including the Exascale Co-design Center for Materials in Extreme Environments, a joint initiative with Los Alamos that has produced CoMD and various other mini-apps.

Mini-apps serve as a common language that researchers and developers can use to understand one another's needs and requirements during the co-design process. They expose hardware developers to simplified examples of codes that the scientific user community wants to execute on new systems. Because the tools are easily modified, both parties can alter them as part of design trade-off discussions and then quickly see how those changes affect performance. Lessons learned from the trade-off analyses are applied not only to the hardware designs but also to the full-scale application codes, thereby ensuring researchers can make the most of next-generation HPC hardware when it arrives.

Mini-apps provide a dynamic and increasingly popular platform for HPC testing and co-design activities. They have already aided the development of Livermore's Sequoia and Sandia's Cielo supercomputers and are an integral part of the co-design strategy for DOE's Extreme-Scale Computing Effort. Notes Richards, "Livermore has a long history of cooperation with hardware vendors, but mini-apps are taking it to a new level."

—Rose Hansen

Key Words: application-driven co-design, Co-designed Molecular Dynamics (CoMD), Extreme-Scale Computing Effort, Exascale Co-design Center for Materials in Extreme Environments, high-performance computing (HPC), Mantevo Suite 1.0, mini-app, R&D 100 Award, Sequoia.

For further information contact James Belak (925) 422-6061 (belak1@llnl.gov) or David Richards (925) 424-5140 (richards12@llnl.gov).